Network Provisioning & Resource Management for IP Telephony[1]

Chen-Nee Chuah and Randy H. Katz
{chuah, randy}@cs.berkeley.edu
Department of Electrical Engineering and Computer Science
University of California at Berkeley

ABSTRACT

The rapid growth of IP-based packet switched network and the overall bandwidth efficiency of an integrated IP network make it an attractive candidate to transport real time voice connections. However, high quality voice over IP (VoIP) remains a challenge because interactive voice imposes many performance requirements (such as loss rate and latency) on the transport network. This cannot be easily achieved in the current Internet's "best-effort" transport. In this paper, we investigate the aggregate behavior of multiplexed voice flows and try to leverage the statistical multiplexing property to design a better network. With proper network provisioning, it is possible to carry VoIP traffic with excellent voice quality without serious under-utilization of resources in a well-managed IP network. We review VoIP architecture over Differentiated-Service model, and analyze the performance using both analysis and simulation. We show that knowledge about the mean m and variance σ^2 of the individual voice source is sufficient to estimate the bandwidth usage, C_v, of the aggregate voice traffic under specific performance requirements. We estimate the bandwidth usage in the form of $C_v = N \cdot m + k \cdot \sqrt{N} \cdot \sigma$ (where N = number of users) and investigate how k captures the multiplexing gain as well as the specific loss rate required. The results are useful for making such decisions as choosing appropriate output link bandwidth or proper bandwidth allocation for VoIP traffic in VPNs. Our experiments show that if we leverage the statistical multiplexing property, we can admit more than twice the number of voice sources compared with allocating the peak rate to each flow. We propose a sender-assisted call admission control policy for Voice VPNs to handle voice set up requests based on our results.

[1] This study was supported in part by DARPA under Grant No. N66001-96-C-8508, Lucent Technologies, and Ericsson.

1 INTRODUCTION

The proliferation of portable computers, hand-held digital personal communicators and smart multimedia end devices have resulted in an explosive growth of Internet traffic. As a result, the packet-switched wide-area network infrastructure is growing rapidly to accommodate the demand of IP-based end devices to access the backbone, which also makes it an attractive candidate for a backbone supporting voice and data applications. In fact, multiplexing data and voice results in a better bandwidth utilization than the traditional circuit-switched voice-or-nothing backbone in the PSTN (Public Switched Telephone Networks) consisting of over-engineered voice trunks. This justifies looking at voice as service on future Internet packet networks such as ISPN (Next Generation Internet Integrated Services Packet Network). In fact, the advent of audio/video conferencing tool such as Netmeeting, NeVot, vic & vat, Rat, CuSeeMe, and so forth make real-time video/audio streams as well as delay sensitive packet audio an increasing fraction of future Internet traffic.

However, the Internet is designed for “best effort” datagram service with no assurance for actual packet delivery. Since there is no dedicated end-to-end connection/links between the sender and receiver, packet loss, out-of-order delivery, delay jitter and latency are bound to occur when the shared network is congested. This has significant adverse impact on the perceived quality of real-time applications. In this paper, we consider voice service on the Internet, also known as *Voice over IP (VoIP)* or *IP Telephony*, as the primary target application because its subjective properties in the presence of delay/packet loss and its traffic characteristics are relatively well-known. Before the widespread business deployment of Internet Telephony is possible, advances is needed in protocol standards, router/switch capacity, elaborate packet classifications and Quality of Service (QoS) management techniques to provide toll quality interactive voice communication.

1.1 INTERNET QOS

There are currently two approaches to enhance QoS for real-time flows such as VoIP. The first relies on application-level QoS mechanisms to improve QoS without making changes to the network infrastructure. For instance, RTP/RTCP transport protocol[1] is designed to transport services for real-time applications. Voice data is transmitted using UDP packets (RTP) to the receiver, while the RTCP packets provide feedback on delay, jitter and losses to the sender. The application then changes codec parameters such as sample sizes or output rate to adapt to the network condition based on the statistics in the RTCP packets. Other application-level QoS mechanisms include layered encoding, low-bit-rate coder[2], forward error correction (FEC)[3], and other source-adaptive mechanism.

The second approach relies on the network-level QoS mechanism to provide variable grade of service with performance guarantees to the heterogeneous mix of internet flows. IETF proposed two models to provide Internet QoS: Integrated Services (Int Serv)[4] and Differentiated Services (Diff-Serv)[5][6]. The philosophy behind Int Serv is that routers must be able to reserve resources for individual flows[1] to provide QoS guarantees to end users. Unfortunately, the amount of per flow state information that needs to be kept at routers increases proportionally with the number of flows. This incurs huge storage and processing overhead at routers, and therefore does not scale well in the Internet core backbone. Diff-Serv, on the other hand aggregate multiple flows with similar traffic and performance characteristics into a few classes. Therefore, the backbone routers only need to provide per-hop differential treatments to a few classes of service. This approach requires either end-user applications, first hop routers or Ingress routers (interface where packets enter the core backbone) to mark the individual packets to indicate different service class.

[1] A flow is a sequence of packets from the sender to the receiver but does not necessarily follow the same route.

Currently, this is supported by the 3-bit Type of Service (TOS) field in IPV4 header, which can be used to indicate the need for low delay, high throughput, or low-loss-rate service. Although flow aggregation improves scalability, it becomes unclear what level of statistical guarantees can Diff-Serv provide to individual flows, and if there exists such "guarantees" at all.

A detailed comparison between Int-Serv and Diff-Serv is out of scope of this paper. We believe that a combination of these two models with carefully provisioned networks constitutes a promising solution. In other words, resource reservations for individual flows can be made in fringe networks or virtual private networks where customers or service providers have a prior knowledge of the network topology, and relatively good control over the intermediate routers. On the other hand, a Diff-Serv model will be more appropriate at the Ingress/Egress routers that connect the various types of access networks to the Internet backbone.

1.2 MOTIVATION & GOALS OF THIS PAPER

In this paper, we assume that the Internet backbone network will continue to scale by adding high speed links or switches from time to time to cope with the increasing traffic load. On the other hand, the growth in capacity in the various access networks is slow compared to the growth in needs of the data end points. Although new high speed access technologies like Hybrid Fiber Coax(HFC), Fiber To The Curb (FTTC), Fiber To The Home (FTTH), and various types of Digital Subscriber Loops (xDSLs) are being deployed, we assume that for foreseeable future, bandwidth will remain a scarce resource in the access networks. The scarcity of link bandwidth can sometimes be due to physical layer limitations, e.g., capacity in wireless channels which are subject to interference, fadings and noise are bounded by Shannon Limit. Therefore network provisioning and QoS mechanisms are crucial to ensure that the performance requirement of the end users are satisfied, while making the most efficient utilization of the network resources.

Our studies focus on how to do resource provisioning and admission control in the access network boundary routers, in particular at the congested nodes where packets are put in a queue, so that the end-to-end performance of delay-sensitive applications such as VoIP can be guaranteed. We assume a Differentiated Service architecture which relies on "marking" of individual packets to provide differential treatment of flows on a *per-packet* and *per-hop* basis. We consider VoIP as our target application and rely on either sender or the first hop router to mark the packets differently (e.g., by setting the TOS field of IP header) to indicate whether it is voice, data or other. We will show that it is possible support high quality voice on a managed IP network (voice VPNs or multi-service VPNs[7]) if appropriate protocols and network provisioning are implemented. We are interested in *managed IP backbone* where statistical guarantees are possible rather than voice over intractable public Internet.

First, we investigate how a prior knowledge about the first and second order statistics about the voice sources can help in network provisioning. For instance, virtual private network service providers can make appropriate pre-allocation of bandwidth to different classes of traffic, e.g., delay sensitive vs. best effort, based on off-line measurements of traffic characteristics to make sure that the Service Level Agreements (SLAs) of the customers are satisfied. Next, we study how real-time estimates of traffic statistics could be used to do dynamic resource allocations for VoIP at intermediate routers. In particular, we investigate the performance of estimation based resource allocation with Weighted Fair Queueing(WFQ)[8]. The following are the fundamental network design questions we try to address:

- Given that *n* potential voice sources will arrive at a node *i*, how much bandwidth & buffer space should be allocated to the voice traffic class to guarantee performance in terms of packet loss or delay?

- Given a fixed link capacity, C, what is the maximum number of flows, $n_{\max}$, can we admit into the queue before we run into over-booking problem?
- What kind of statistical guarantees can we provide to each individual flow?
- How do we design the admission control policy and dynamic bandwidth allocation for a core-router that has two queues: one for voice, and one for data (in a Diff Serv environment)?

We approach these issues using both analytical and simulation techniques. Although our study is based on VoIP, the results can be easily extendable to other delay sensitive applications.

This paper is organized as follows: Section 2 gives an overview of the architecture and queueing disciplines that we consider for VoIP solutions. In Section 3, we describes how the voice traffic is modeled, and how we choose the performance requirements based on the effect of packet loss and latency on perceived quality. Section 4 we analyze the network provisioning problem using mathematical model. Simulation results are presented in Section 5 with detailed discussions. We propose a sender assisted admission control policy in Section 6. This is followed by conclusions in Section 7.

2 OVERVIEW

The integration of high quality voice and traditional data services is not easy because the public Internet consists of a large number of loosely coupled subnetworks, and therefore the delivery of packets are hardly predictable. For our study, we assume Diff-Serv Model for the IP core network to provide differential QoS to different end-to-end applications.

2.1 VOICE OVER DIFFERENTIATED SERVICES

The basic idea behind Diff-Serv model is to mark the packets to indicate whether or not the packet should receive preferential treatment. One example is the Type of Service (TOS) byte in the IPv4 header to indicate the need for low-delay, high-throughput or low-loss-rate service. A set of packet forwarding treatments (per hop behaviors, or PHB) is described in [9]. Since there are only a limited number of service classes indicated by the TOS field, the number of state information is proportional to the number of classes rather than number of flows, and therefore Diff-Serv approach is more scalable.

An end-to-end service architecture has been proposed in [10] and analyzed in [11]. It provides *assured service* and *premium service* in addition to best-effort service. *Assured service* is intended for customers that need reliable services from service providers. A Service Level Agreement (SLA) is some sort of a contract between the two parties that specifies the amount of bandwidth allocated for the customers. The customers themselves are responsible for deciding how their applications share that amount of bandwidth. *Assured service* can be implemented by sending all packets to an *Assured Queue* managed by random early detection (RED)[12][13] with In and Out - RIO[14]. *Premium Service* on the other hand provides low-delay and low-jitter service. The SLA specifies a fixed peak bit rate, which the customer is responsible for not exceeding. All excess traffic are dropped. *Premium service* is suitable for Internet telephony, video-conferencing and for creating virtual lease lines for Virtual Private Networks (VPNs). For better link utilization, dynamic SLAs should be supported so customers can request bandwidth on demand. However, admission control policy is needed so that the shared link is not **over subscribed**.

In this paper, we assume that voice packets are marked as *P* (Premium service) while data packets are marked *BE*, and forwarded in best-effort manner. We rely on either the sender of first hop router to mark these packets appropriately. All packets with the P-bit set enter a *premium queue* (PQ) while data packets enter a *best-effort queue* (BEQ). Packets in PQ are always forwarded before packets of other classes, and hence in the extreme case, they can potentially use 100% of the output link bandwidth. To prevent starvation of BEQ traffic (such as TCP with congestion control), we fix a minimum data throughput, W_D, that must be sustained. This ensure that a minimum bandwidth is allocated to transport packets in BEQ even in times of network congestion. However, this comes with a trade-off in terms of the maximum number of PQ flows that can be admitted. Another approach is to use Weighted Fair Queueing (WFQ) to serve PQ and BEQ at a different rate depending on the weights assigned to the different queues.

2.1.1 NETWORK COMPONENTS

In a Diff-Serv network architecture, certain functionality is required from the boundary routers which reside at the ingress and egress points to and from the diff serv IP core network. Classification, policing, if necessary, shaping can be done at the ingress router according to the rules derived from the SLAs. We assume all VoIP sources are relatively well-behaved and never exceed the peak rate specified in the SLAs. Therefore policing and shaping are not addressed in this paper. We consider a very simple architecture as shown in that Figure 1 includes two important functional blocks: IP Telephony Gateways (IPTG) and Diff-Serv Boundary Routers (DSBR).

IPTGs performs the necessary conversion between the transmission format of the input voice traffic and RTP/UDP/IP format that are carried over Diff-Serv networks at the ingress and egress points. The origin of voice traffic can be PSTN phones, cellular phones in digital wireless access networks (GSM, CDMA etc.) or even multimedia applications such as Vat, NetMeeting from Local Area Networks(LANs). Each of them can use different voice codecs and transmission formats. Therefore, we assume that IPTGs perform some or all of the following functions when necessary to convert input voice traffic to RTP packets:

- coding (PCM, ADPCM, DVI, GSM format etc.)
- silence suppression
- packetization (collect enough voice frames to form a packet), and
- convert into RTP/RTCP packets with the TOS bytes properly marked in IP header.

We assume that voice traffic that arrives at the boundary routers are in the format of properly marked RTP packets at a constant rate when the voice source is active and zero packet transmission during silence periods.

To provide differential treatment to packets from different classes of service, in our case voice vs. data, DSBR need to implement the following components (refer to Figure 3 & discussion in Section 4),

- A classifier: Classification is a necessary function for a router/switch that treats certain traffic differently from the other. It can be implemented in various degrees of granularity (for each source-destination pair, source-destination address and port number, or for a specific service class). For our studies, the classifier can differentiate voice and data traffic based on TOS field and separate them into two different queues (PQ & BEQ).
- A scheduler: Scheduling policy is part of queue management which decides which packet to transmit next. It can choose from a single queue (e.g., FIFO) or from multiple queues (e.g., SFQ, WFQ, CBQ, CBQ with weighted round robin).
- Buffer manager: The buffer management scheme is responsible for putting packets in queue as they arrive and decide which packets to drop when buffer memory is exceeded. Examples include drop-tail, RED, RIO.

For our studies, we assume that WFQ is deployed at the outgoing link of DSBR to serve PQ and BEQ, both are a simple Drop-Tail buffer and are served on a first in first out (FIFO) basis.

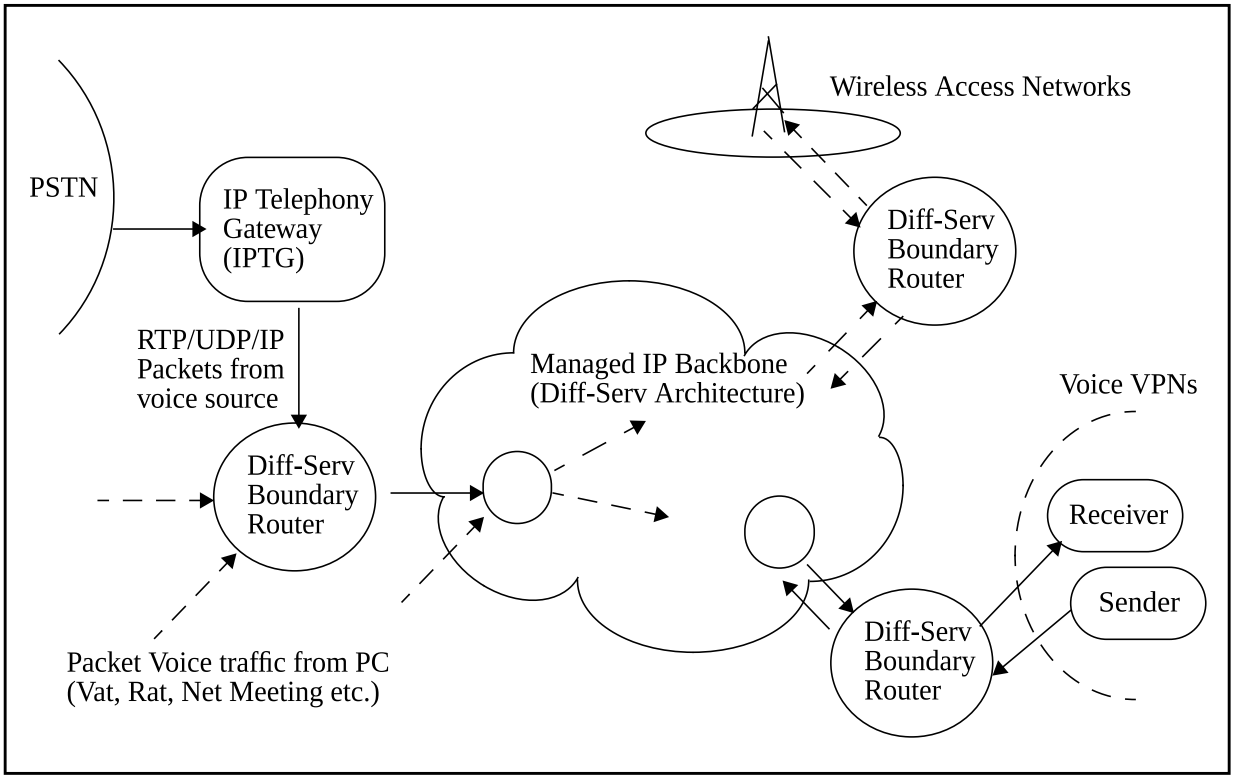

Wireless Access Networks
PSTN
IP Telephony Gateway (IPTG)
Diff-Serv Boundary Router
RTP/UDP/IP Packets from voice source
Managed IP Backbone (Diff-Serv Architecture)
Voice VPNs
Diff-Serv Boundary Router
Receiver
Sender
Diff-Serv Boundary Router
Packet Voice traffic from PC (Vat, Rat, Net Meeting etc.)

Figure 1. An end-to-end service architecture for Voice over IP using Differentiated Services.

2.2 TRANSPORT PROTOCOL

We assume that the voice samples are packetized and carried over RTP/UDP/IP protocol stack. RTP[1] is a general-purpose real-time data streaming protocol that provides intra- and inter-media payload synchronization, payload identification and sequencing. RTCP is the transport control protocol that enable application to send/receive traffic statistics, jitter and loss estimates. Since we are only interested in designing network-level QoS mechanism, we assume that the IP telephony applications do not perform any kind of congestion control in terms of rate adaptation. If the packet loss and delay jitter estimates are obtained from RTCP sender/receiver reports, the applications can choose to tolerate the degradation in QoS or terminate the communication. This is equivalent to “dropping” a call in PSTN terminology.

3 VOICE OVER IP

Voice over IP (VoIP), refers to real-time delivery of packet voice across networks using the Internet protocols. Most recent research on VoIP addresses issues involved with PC to PC and PC to PSTN or cellular phone voice communications.

3.1 TRAFFIC MODEL

The IPTG shown in Figure 1 provides conversion between different transmission formats and RTP/UDP/IP protocol stack. With silence suppression, voice traffic that enter the boundary routers can be modeled as an on-off (or birth-death) Markov processes. For each individual VoIP packet flow, the alternating periods of activity and silence are exponentially distributed with average durations of $1/\mu$ and $1/\lambda$, respectively. The fraction of time that the voice source is "on" is $\frac{\lambda}{\lambda+\mu}$. We assume that when the source is in the "on" state, fixed-size packets are generated at a constant interval. On the other hand, no packets are transmitted when the source is "off". The size of the packet and the rate at which the packet is sent depend on the voice codecs.

Traditionally voice is Pulse Code Modulated (PCM, G.711) at 64kbps in the PSTN. PCM provides high quality reproduction of speech and comparable quality can be maintained with ADPCM. Recent advances in compression technology have allowed highly compressed speech (16kbps and lower) that offer excellent voice quality in absence of packet losses.

3.2 PERFORMANCE REQUIREMENTS

High quality interactive voice imposes many performance requirements on the underlying transport network. For example, one way end-to-end delay should be under 150 ms to preserve the quality of interactive communication. In a circuit switched network, propagation delay is the only significant component in the one way end-to-end delay. In addition, this delay is constant component during the entire call duration, and therefore can be easily controlled. VoIP architecture, on the other hand, introduces new delay components such as: coding/decoding delay, packetization delay, queuing delays at intermediate routers/switches, and jitter compensation delay introduced by playout buffers. The multiplexing of VoIP and data traffic on shared links also introduces packet losses caused by buffer overflow at congested nodes. Latency and packet losses have adverse impact on the perceived voice quality, and therefore need to be bounded.

Our goal is to show how high quality voice can be supported with maximum utilization of resources if the network resource is provisioned properly and distributed admission control is implemented. To achieve this, we need to quantify the performance of VoIP. Proper network provisioning and resource allocation bound the maximum queueing delay and packet loss rate so that statistical guarantees can be given to individual voice source. Therefore we need to map both delay and packet loss to perceived voice quality.

3.2.1 MAXIMUM TOLERABLE DELAY

In this paper, we ignore the delay introduced by the playout buffer. We also assume that one propagation delay is relatively constant and can be easily estimated. We assume the sender uses the same codec throughout the call duration, and chooses a fixed sampling rate and packet size at the beginning of the call. Since we are interested in investigating the effect of bandwidth allocation on voice quality, we try to segregate the effects of application-level QoS mechanisms. We assume that no application-level congestion control or rate adaptation are deployed at the voice sources. The only highly variable delay component in our model is queueing delay that occurs due to the multiplexing of voice packets, as well as the integration of voice and data over a shared link.

ITU-T Recommendation G. 114[15] specifies that one-way transmission time for connections with adequately controlled echo should be in 0-150 ms range to be acceptable for most user applications. As mentioned earlier, the end-to-end delay for VoIP depends on various components of the packet network. The total delay from these components must be smaller than 150 ms. Since queueing delay is the only variable part in our model, we try to budget the per hop queuing delay when we design the resource allocation schemes and address network provisioning issues in the next section.

In our experiments, we assume PCM transcoding introduces almost negligible delay if implemented in hardware(0.75 ms). From [15], Public Land Mobile Systems contribute around 80 - 110 ms to one-way propagation time. Satellite systems introduce 12 ms at 1400 km altitude, and 110 ms at 14,000 km altitude. Optical fibre cable system contributes around 50-60 ms from coast to coast in United States. Let us assume that it takes 100 ms propagation delay for voice packets to be transported across United States. Hence the total queueing delay should be kept within 50 ms (150ms - propagation delay). From traceroute, we found out that there were typically around 8-12 hops between a machine on the west coast and the east coast. Assuming that queueing delay is almost the same for each hop, we must keep the per hop queueing delay to be **at most 5 ms**.

3.2.2 PACKET LOSS RATE

Packet losses can cause further distortion beyond the unavoidable loss of information introduced by speech encoding/decoding and therefore should be minimized. We consider packet losses that are caused by buffer overflows in routers as well as discarding of delayed packets in the receiver playout buffer (i.e., if packets arrive at the receiver after too long a delay and miss the playout time, these packets are discarded and therefore considered lost). The impact of losses on voice quality depends on the loss ratio, burstiness of losses, frame sizes per packet and the voice codec. The impact of packet loss on voice quality is dependent on the voice codec used. We use *vat*[1] to run a simple subjective test to map the packet loss rate to perceived voice quality for the following case: Using PCM codec with silence suppression, 8 kHz sampling rate, 8 bits per sample (contributing to 64kbps when the source is active), and 20 ms of voice samples per packet.

The sound files of three sentences (about 6 seconds each) from the movie, “A Few Good Men” are downloaded[2] and converted to PCM format with 8 kHz sampling rate. The voice samples are packetized into RTP packets with 12-byte RTP Header and sent through a simple network emulation that introduces uniformly distributed packet losses according to different loss rates. The perceived voice quality is scored on a numeric 0 to 5 scales with the following definitions: 5 = crystal clear, 4 = sentence is comprehensible but less clear; 3 = speech become choppy; 2 = sentence gets harder to comprehend due to noise; 1 = can comprehend less than 50% of the sentence; 0 = gibberish noise. The result is plotted in Figure 2. Results show that the tolerable loss rates are within **1-3%** and the quality becomes intolerable when more than 3% the voice packets are lost. Note that packet voice using forward Forward Error Correction (FEC) is more resilient to losses and therefore we expect the curve to shift to the right in this case. On the other hand, the quality of voice connection using compressed speech is more sensitive to lost voice samples (left shift of the curve in Figure 2). Also, bursty losses can impair the voice quality much worse than uniform losses.

[1] Vat is an Audio Conferencing tool developed by Network Research Group of Lawrence Berkeley National Laboratory (http://www-nrg.ee.lbl.gov/vat/)

[2] Since these sound files are in WAV format, we use sndrfmt program from ICSI to resample the voice at 8KHz and convert the format to PCM and saved as μ-law bytes.

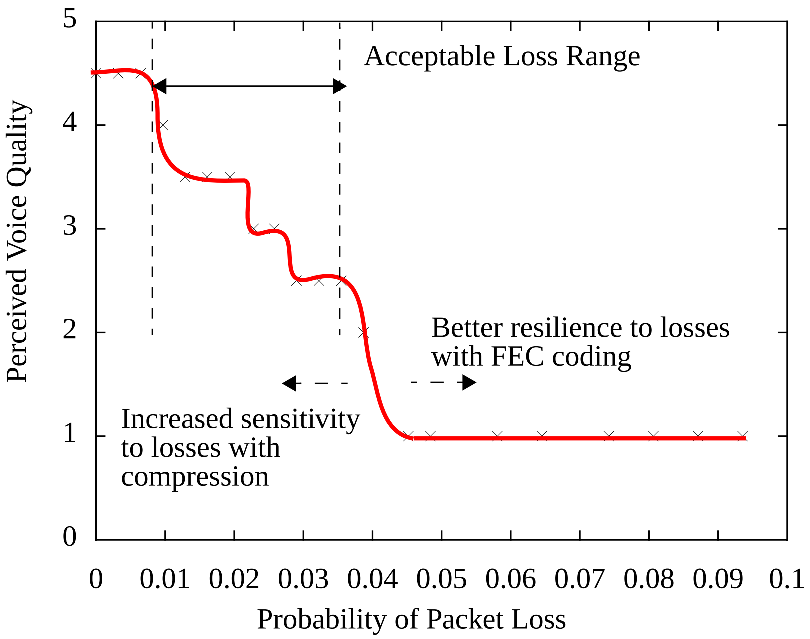


Figure 2. Perceived voice quality when different packets are dropped randomly according to different loss rates.

4 NETWORK PROVISIONING FOR VOIP

The Diff-Serv Boundary Routers (DSBR) shown in Figure 1 are placed at the ingress and egress points where the packets enter or leave the Diff-Serv IP backbone. They can be the edge routers for a privately owned and managed networks (Virtual Private Networks), or interface between LANs or non-conventional access networks (Wireless LANs, PSTNs) to access the IP backbone. In our analysis, we focus on the DSBRs that connect managed access networks such as Voice VPNs or multi-service VPNs to the backbone, and we only consider two types of traffic: voice and data. We assume a DSBR containing a classifier, a WFQ scheduler and FIFO Drop Tail queue manager as shown in Figure 3. The classifier will differentiate the voice packets from data packets and separate them into two different queues: PQ and BEQ, respectively. These two queues are served in rotation by a WFQ by allowing some maximum number of bytes from one service class to be served each time based on the weights, w_{v} & w_{d}, that are assigned to PQ (voice) and BEQ (data). The challenge is to decide how much capacity, C, should the output link be, and how to provision the resources for VoIP (w_{v} & w_{d}) to make sure that the performance requirements to achieve high perceived voice quality are satisfied.

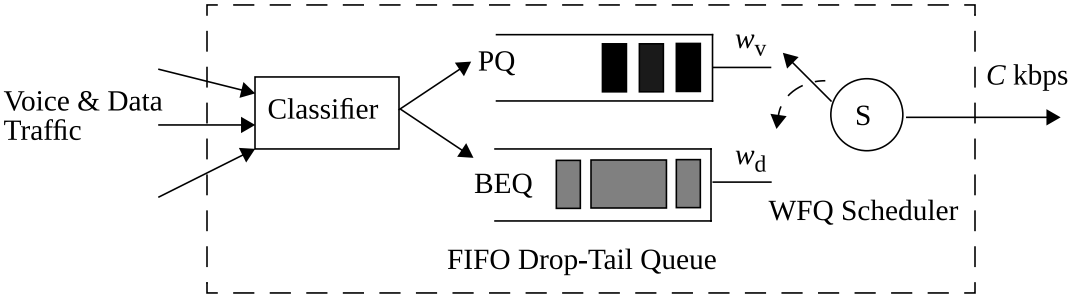


Figure 3. Functional blocks of a boundary router connecting VPNs or local access networks to the managed IP backbone.

We are particularly interested in addressing the following two scenarios:

- Consider voice VPNs where the only type of traffic is packet voice. It is reasonable to assume that the number of possible voice sources, *N*, is known or can be easily estimated in a managed VPN. For instance, *N* can be equal to the number of phones plus the number of PCs with multimedia applications. Then there is only one queue (PQ) in the DSBR for this type of network ($w_v = 1, w_d = 0$), and the question becomes how much physical bandwidth, *C*, is needed at the output link to preserve the interactive feeling and perceived quality of all the voice calls transported over this link.
- Consider a multi-service VPNs where two types of traffic, voice and data, share the output link. The question becomes how do we pick w_v & w_d so that sufficient bandwidth is allocated to VoIP while allowing a minimum throughput of data to be transmitted and maintaining high utilization of the link through bandwidth sharing.

4.1 ANALYTICAL MODEL

In this section, we present an analytical model to demonstrate that proper resource provisioning is sufficient to meet QoS constraints for VoIP. We consider a fluid model of PQ (queue for voice packets) alone, with *N* potential incoming voice traffic and served at C_v kbps, where $C_v = \frac{w_v}{w_v + w_d} \cdot C$. We assume silence suppression is deployed at the application level. Each voice source is modeled as an on-off Markov process as in Section 3.1. Let $X_i(t)$ be the instantaneous rate of voice connection *i*, then,

$$X_i(t) = \begin{cases} \mathrm{R} & \text{when the voice source is active} \\ 0 & \text{when the voice source is silent,} \end{cases} \tag{1}$$

where R is the voice bit rate determined by the codec and compression scheme deployed at the source. The rate of transition from state ‘0’ to state ‘R kbps is λ while the reverse transition happens at the rate of μ. For our analysis we assume that the random processes $X_i(t)$ are i. i. d. (independent & identically distributed). The total rate of the aggregate traffic is *Y*(*t*):

$$Y(t) = \sum_{i=1}^{N_A} X_i(t) \tag{2}$$

where N_A is the number of actual voice calls in progress. The maximum possible value of N_A is N, which is the maximum number of possible voice sources in the network. For example, N in a VPN can be the total number of telephone handsets or other end devices that are capable of generating voice traffic.

At any particular time instant, say $t = \mathrm{T}$, $X_i(\mathrm{T})$ are just discrete time random variables. We assume that the random process $X_i(t)$ is ergodic, i.e., time averages see the ensemble averages, and stationary, i.e., $X_i(\mathrm{T})$ have the same statistics at any time instant T. Since $Y(t)$ is the sum of i. i. d. stationary processes, $Y(t)$ is also stationary. At any time instant, $t = \mathrm{T}$, $Y(\mathrm{T})$ is just the same of i. i. d. random variables. For simplicity, we omit the time dependence and use the notations X_i, Y instead. We assume the stationary distribution of X_i is:

$$P(X_i = x) = \begin{cases} \dfrac{\lambda}{\lambda+\mu} & \text{when } x = \mathrm{R} \\ \dfrac{\mu}{\lambda+\mu} & \text{when } x = 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

We are interested in estimating the minimum capacity C_v that need to be allocated to voice traffic (PQ) so that the loss rate per flow is less than δ. Assume scheduler/server is work conserving and non-preemptive. If we assume that the queues are bufferless, then losses occur when the sum of arrival rates is greater than the rate at which the queue is served: $Y > C_v$. We make the following approximations:

- We consider the worst case where all the potential voice users have calls in progress, i.e., $N_A = N$.
- As soon as the aggregate rate Y exceeds the server rate, information (in bits) from some connections are lost. Let say this happens with probability $P(Y > C_v) = \delta$. The losses can be shared by some connections, or in the worst case, the losses may happen to only one connection. We assume that in any cases when losses happen, the worst loss rate that is suffered by an individual voice flow is not more than the total loss rate, δ.
- To achieve satisfactory voice quality, loss rate per source has to be bounded: $\delta \le \delta_{max}$.

Therefore, we have the following worst case constraint:

$$P(Y > C_v) \le \delta_{max} \tag{4a}$$

$$\Rightarrow P\left(\sum_{i=1}^{N} X_i > C_v\right) \le \delta_{max} \tag{4b}$$

When N gets large, Y tends to have a normal (Gaussian) distribution under the *Central Limit Theorem*[16]. In short, the Central Limit Theorem states that the sum of a large number of independent observations from any distribution tends to have a normal distribution, and this is true for observations from all distributions. Given that X_i's are i. i. d. with mean m and variance σ^2 (this easily can be determined from Eq. (3)), and Y is the sum of X_i's, the mean and variance of Y is simply the sum of the mean and the sum of the variance of X_i's, respectively. Therefore Y is normal distributed with mean Nm, and variance $N\sigma^2$,. $Y \sim N(Nm, N\sigma^2)$. To solve Eq. (4a), we can use the well-known Q-function:

$$\begin{aligned} P(Y > C_{\mathrm{v}}) \leq \delta_{\max} &\Rightarrow P\left(\frac{Y - Nm}{\sqrt{N} \cdot \sigma} > \frac{C_{\mathrm{v}} - Nm}{\sqrt{N} \cdot \sigma}\right) \leq \delta_{\max} \qquad (5) \\ &\Rightarrow P\left(Z > \frac{C_{\mathrm{v}} - Nm}{\sqrt{N} \cdot \sigma}\right) \leq \delta_{\max} \\ &\Rightarrow P\left(Z \geq \frac{C_{\mathrm{v}} - Nm}{\sqrt{N} \cdot \sigma}\right) \leq \delta_{\max} \qquad (\text{since Z is continuous}) \\ &\Rightarrow Q\left(\frac{C_{\mathrm{v}} - Nm}{\sqrt{N} \cdot \sigma}\right) \leq \delta_{\max}^{\cdot}, \end{aligned}$$

where Z is a normalized zero mean unit variance normal random variable, and

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{\left(\frac{-t^2}{2}\right)} dt \qquad (6)$$

Using the inverse Q-table, we can determine the value of C_{v} for any given $\delta_{\max}$:

$$C_v \geq Nm + Q^{-1}(\delta_{\max}) \cdot \sqrt{N} \cdot \sigma \qquad (7a)$$

4.2 APPLICATION OF RESULTS

In practice, the voice traffic arrives in IP-packet format, and there are buffers in the system. The ideal fluid model in Section 4.1 does not hold, but we can still use the results in Eq. (7a) as a first order approximation of the actual bandwidth required. From Figure 2 in Section 3.2, the packet loss rate should not exceed 3% to preserve satisfactory voice quality. Therefore, we choose $\delta_{\max} = 0.03$. With $Q^{-1}(0.03) = 1.88$, we need to allocate bandwidth of at least:

$$C_v = N \cdot m + 1.88 \cdot \sqrt{N} \cdot \sigma \qquad (7b)$$

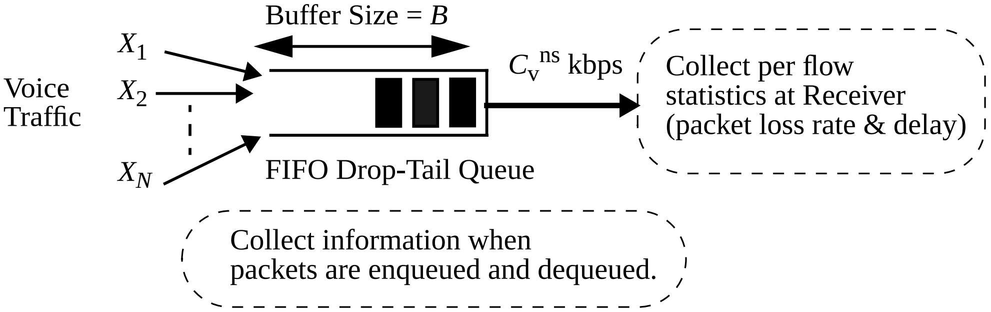
to VoIP. As a numerical example, let $\lambda = 0.4$, $\mu = 0.6$, R = 80 kbps. From Eq. (3), the mean $m = 0.4*80 = 32$ kbps, while $\sigma = \sqrt{E[|X - m|^2]} \approx 39.2$ kbps. For every value of N, we can then estimate the capacity needed C_{v} by substituting the numerical values into Eq. (7b).

Since there is buffer in the queue, the packet loss will be less in practice. In addition, there is a statistical multiplexing gain as more and more flows are aggregated together, which results in a lower "effective" bandwidth per flow. In other words, the aggregate mean rate and variance of Y can be less than the sum of individual mean and variance due to the nice multiplexing property that is not captured in our analysis above. As a result, the actual bandwidth that is needed to meet the performance requirement of VoIP can be less than the predicted value of C_{v} from Eq. (7b), which can be viewed as a upper bound. For comparison purposes, we denote the required bandwidth computed from analytical model as $C_{\mathrm{v}}^{\mathrm{o}}$, and compare it with the actual bandwidth determined from simulation, denoted as $C_{\mathrm{v}}^{\mathrm{ns}}$, in the next section.

5 NUMERICAL RESULTS & DISCUSSIONS

In this section, we use simulation to determine the minimum bandwidth, C_v^{ns}, that is needed to support a specific number of voice users, N, and investigate how well does Eq. (7b) predict the bandwidth requirement. We also study the effect of statistical multiplexing on bandwidth requirement of aggregate voice traffic.

We use ns simulator[1] to model a simple one-hop topology as shown in. *ns* is a discrete event simulator derived from REAL simulator[17]. The ns development effort is now part of the ongoing VINT[2] project. Ns provides substantial support for simulation of TCP, real time transport protocols, routing and multicast protocols.



Buffer Size = B
X_1
X_2
X_N
Voice Traffic
C_v^{ns} kbps
Collect per flow statistics at Receiver (packet loss rate & delay)
FIFO Drop-Tail Queue
Collect information when packets are enqueued and dequeued.

Figure 4. Simple 1-hop topology in ns.

The voice sources are simulated according to Section 3.1 with voice activity cycle of 40% (i.e., λ=0.4). Assume that 8KHz 8 bits/sample PCM codec is used with 20 ms frame per packet. The voice data packet is 160 Bytes. With 12 byte RTP header, 8 byte UDP header and 20 byte IP header, the size of each packet = 200 Bytes. With these header overheads, the effective rate of a single voice connection when it is active is (200*8)/20 =80 kbps (25% overhead). Buffer size B (number of packets) is chosen such as the maximum possible delay (for packet at the tail of the queue) is at most 5 ms:

$$(B \times 200 \cdot 8) / C_v^{ns} \le 5 \text{ ms.} \qquad (8)$$

We consider N from 0 to 200. For each N, we vary the link bandwidth C_v and observe the packet loss rate. Buffer size B is determined using Eq. (8). As C_v is increased, packet loss rates decreases (see Figure 5 that shows the results for two cases: N=20 and N=30). C_v is increased until the worst per flow loss rate is 3%, and the corresponding value $C_v^{ns}(N)$ is recorded. We repeat this procedure for each N and the result is plotted in Figure 6 together with C_v^o that is predicted using Eq. (7b).

[1] ns source code and documentation can be downloaded from http://www-mash.cs.berkeley.edu/ns/.

[2] Virtual InterNetwork Testbed (VINT) is a collaboration among USC/ISI, Xerox PARC, LBNL, and UCB (http://netweb.usc.edu/vint/.

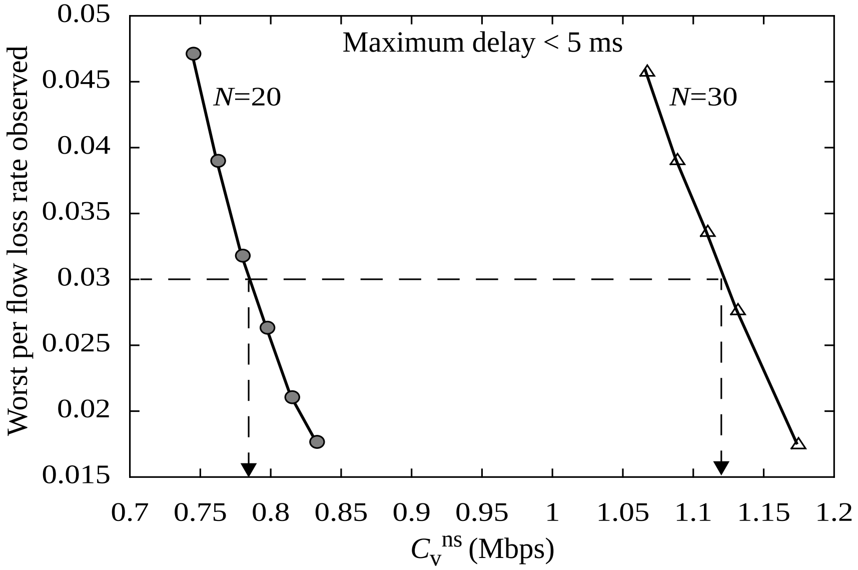


Figure 5. The worst case per flow packet loss rate is plotted against the available bandwidth, C_v^{ns} for different number of users *N*. Two cases: *N*=20 and *N*=30 are shown here. The loss rate decreases as C_v^{ns} is increased. We can determine the required C_v^{ns} to achieve 3% loss rate for each *N* from a family of curves like these, as shown by the dotted lines.

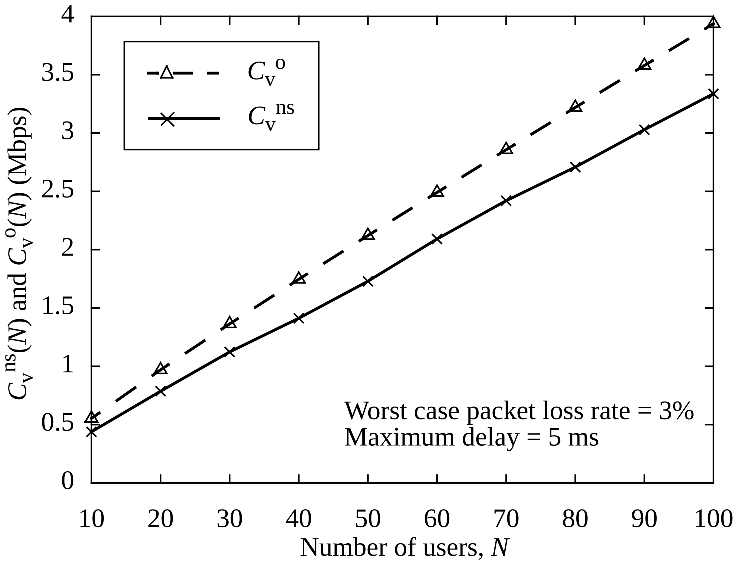


Figure 6. This figure demonstrates that proper bandwidth provisioning is sufficient to support performance requirements of VoIP. C_v^o is the required bandwidth predicted from analytical model while C_v^{ns} is the results obtained from ns simulations.

We observe that the required bandwidth C_v^{ns} increases linearly with *N* at a slope of approximately 33 kbps for each additional voice connection (per unit of *N*). The slope is much less than the peak rate of

80 kbps. In fact, it is very close to the mean rate of 32 kbps, which is roughly 40% of the peak rate. This shows that even for small N, the aggregate traffic of voice flows behaves very nicely and becomes a lot less bursty than the individual source. In fact, the maximum rate of the aggregate traffic rarely goes above its mean rate plus a standard deviation. This is a pleasant discovery since it implies that what we need to know is the mean and variance of each individual source in order to estimate the bandwidth required for VoIP traffic in order to maintain high perceived voice quality. With proper bandwidth provisioning (choosing C_v), voice VPNs and multi-service VPNs can give statistical guarantees (e.g., loss rate) to each individual voice connection.

The same result shown in Figure 6 can be interpreted in a different way. We define N_{max} as the maximum number of users that can be supported so that loss rate is below 3% and delay is less than 5 ms. Using the same experimental settings, we read off the value of N_{max} for each corresponding value of available bandwidth, C_v, from Figure 6 for two cases: (a) observations from ns simulation and (b) predicated value from analytical model. To illustrate the statistical multiplexing gain, N_{max} is plotted against C_v, for these two cases in Figure 7 (solid and dashed lines). For comparison, we consider a third case where we ignore the statistical multiplexing property of the aggregate voice traffic, and simply allocate the peak rate of 80 kbps to each flow (assuming the worst case scenario in which all voice sources become active at the same time). N_{max} in this case is simply the largest integer such as $N_{max} \cdot 80 \le C_v$ kbps N_{max} (dotted line in Figure 7). This line lies much lower than the previous two cases. This implies that the peak rate allocation is over-conservative and resources are under-utilized. For example, at 6.4 Mbps link bandwidth, we can only support 80 users with peak rate allocation. But simulation shows that we can support up to 196 users, 2.4 times as many, while maintaining the 3% loss rate requirement. If we allocate bandwidth based on Eq. (7b), we can support N_{max} = 170, which is still more than double the N_{max} with peak rate allocation.

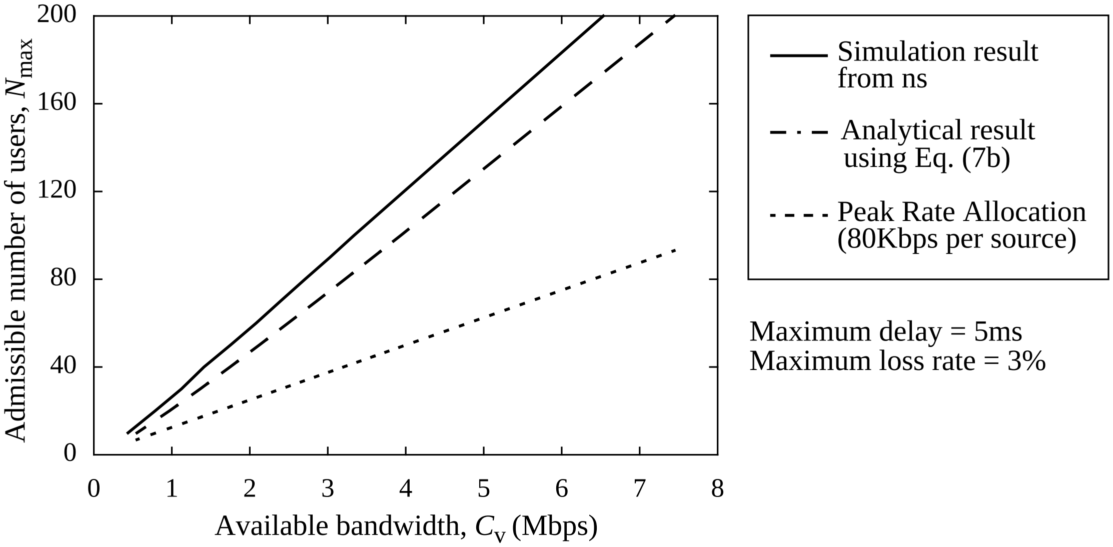


Figure 7. The first two lines (solid and dashed) are the identical to the result shown in Figure 6. The third line (dotted) is obtained by allocating peak rate of 80 kbps to each user.

5.1 ECONOMIES OF SCALES

C_v^o tracked the actual required bandwidth fairly well for moderate *N*. At *N*=100, the gap between C_v^o and C_v^{ns} is 18 %. The observation suggests that although Central limit theorem works well for moderate *N*, the theoretical limit consistently over-estimate the bandwidth required by the aggregate voice traffic. This is because our analysis in Section 4.1 does not capture the economies of scales in the number of flows, which we investigate in this section. We try to relate actual bandwidth required, C_v, to *N* in a similar form as Eq. (7b):

$$C_v = N \cdot m + k \cdot \sqrt{N} \cdot \sigma \qquad (9)$$

where $m = 32$ kbps, and $\sigma = \sqrt{E[|X-m|^2]} \approx 39.2$ kbps. If the aggregate traffic is truly Gaussian, *k*=1.88 as determined in Section 4.1. However, *k*<1.88 in practice because of statistical multiplexing and a simple consequence of the law of large numbers. In fact, we expect *k* to decrease with the increase of *N* because of the economies of scale in the number of multiplexed sources. Our conjecture is confirmed by simulation results by fitting the measured C_v^{ns} into Eq. (9) to determine *k*. Results show that *k* decreases rapidly with the increase of *N* and saturates around 0.2-0.4 as shown in Figure 8(a). This implies that the variance of the aggregate traffic decreases as more and more voice flows are multiplexed together. To further illustrate the economies of scales in the number if multiplexed voice flows, the “effective bandwidth” per flow is plotted in Figure 8(b). “Effective bandwidth” per flow is determined from simulation by dividing C_v^{ns} by *N*. Results show that the effective bandwidth decreases and slowly converges to the mean rate (32 kbps) as the number of flows get large.

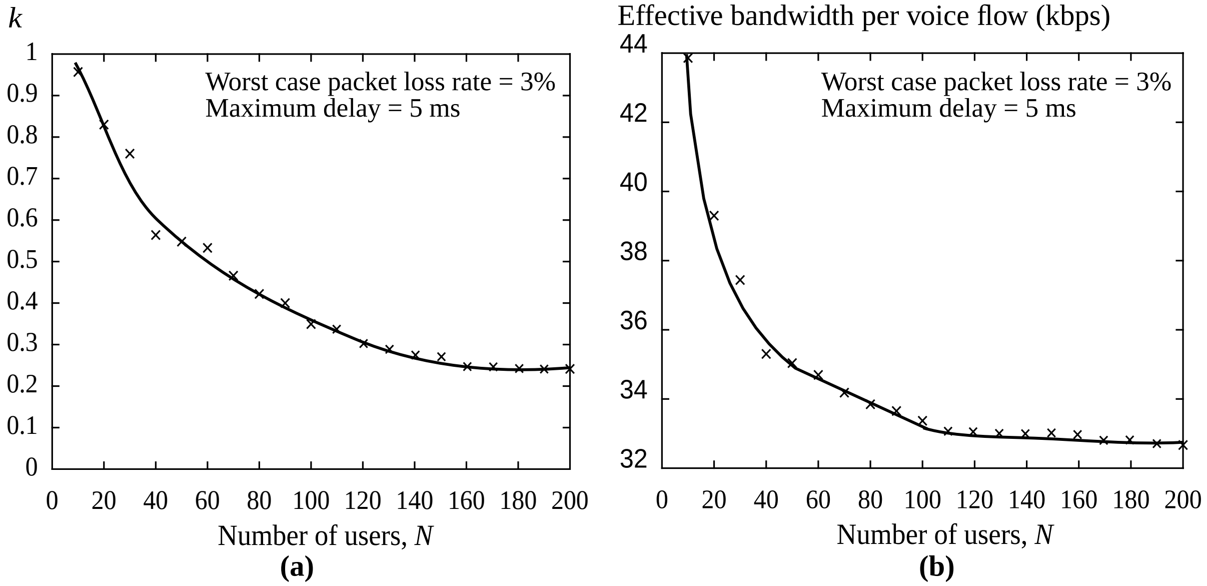


Figure 8. (a) The first graph shows the relationship between the parameter *k* with the number of users *N*. (b) The decreasing effective bandwidth per source in the second graph shows the economies of scales in the number of multiplexed flows.

6 EXAMPLE: SENDER ASSISTED ADMISSION CONTROL POLICY

The results obtained in the previous section can be used as a guideline for dynamic resource provisioning for VoIP at intermediate routers. This section discusses how one can use the results in Figure 6 and Figure 8 to design an admission control policy at routers to provide end-to-end QoS for VoIP.

In general, boundary routers may not know in advance the number of voice sources that can become active at anytime. Admission control policy is essential to ensure that the existing voice connections can be carried over the IP backbone with high quality without being disrupted by the addition of new connections when there is no more idle bandwidth available to support the new flows. On the other hand, we might also want to control the total number of voice flows so that we can maintain a minimum throughput for other classes of traffic, e.g., to avoid starvation problem of TCP connections.

For illustration purposes, we assume that every new voice connection sends out a call set up request that contains a traffic specification (TSPEC) describing the source traffic, e.g., in YESSIR or RSVP. If we assume the TSPEC carries at least the mean and variance (m', σ) of the source, then we can accept/reject the call at the egress boundary routers (where access networks are connected to the IP-backbone) based on the algorithm described in Figure 9. Let there be n existing voice calls when the new call set up request arrives. $m_{\mathrm{est}}(n)$ and $\sigma_{\mathrm{est}}{}^2(n)$ are the measured mean rate & variance of n voice flows at router i. We compute the new mean and variance using the (m', σ'), obtained from the TSPEC, and estimate the bandwidth required, $\tilde{C}_{\mathrm{v}}(n+1)$. We then compare this to the total available bandwidth for VoIP, $C - W_{\mathrm{D}}$, where W_{D} is the minimum bandwidth made available for the combined traffic supported using “best effort” data service. H and p are introduced as hysteresis parameters for admission control purpose. The larger the hysteresis level is, the more stringent the admission control policy, whereby fewer new calls are accepted but better statistical guarantees are given to each of the existing voice calls.

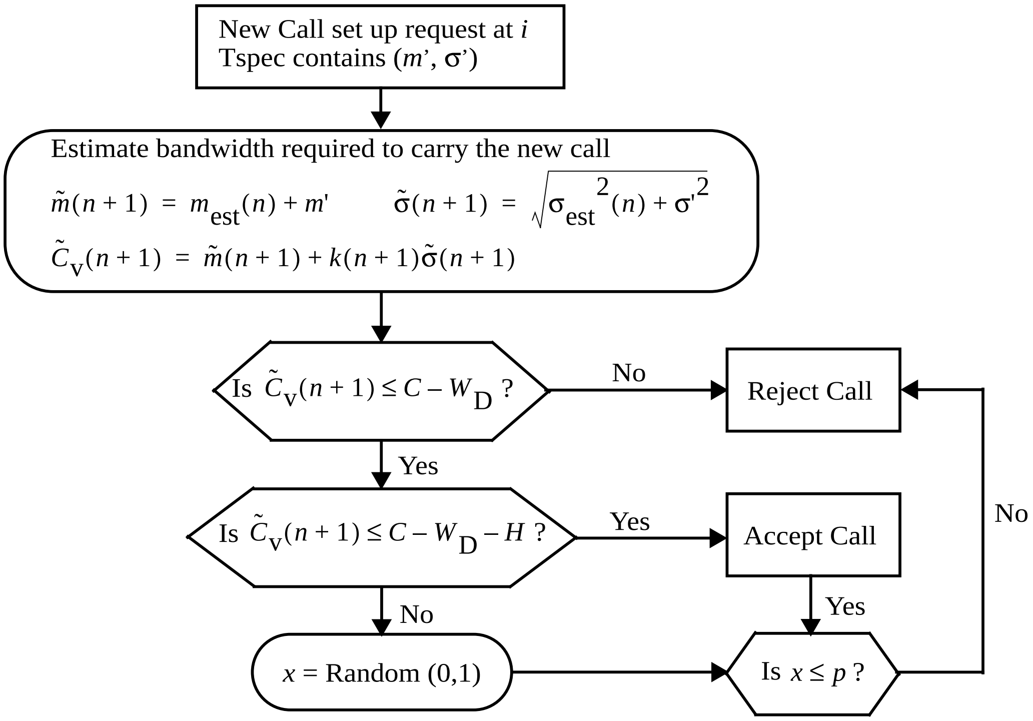

New Call set up request at i
Tspec contains (m', σ')
Estimate bandwidth required to carry the new call
$\tilde{m}(n+1) = m_{est}(n) + m'$
$\tilde{\sigma}(n+1) = \sqrt{\sigma_{est}^2(n) + \sigma'^2}$
$\tilde{C}_V(n+1) = \tilde{m}(n+1) + k(n+1)\tilde{\sigma}(n+1)$
Is $\tilde{C}_V(n+1) \le C - W_D$?
No
Reject Call
Yes
Is $\tilde{C}_V(n+1) \le C - W_D - H$?
Yes
Accept Call
No
No
Yes
x = Random (0,1)
Is $x \le p$?

Figure 9. An algorithm for call admission control for router *i*. A new call arrives when there are *n* existing voice connections.

7 CONCLUSIONS

High quality voice service over IP backbone remains a challenge because the Internet's "best-effort" is not sufficient to support the more sensitive performance requirements of VoIP. However, the overall bandwidth efficiency of an IP network that integrates voice and data traffic together is very favorable. With proper understanding of how the aggregate voice traffic behaves, we can provision the network for VoIP to satisfy all the performance requirements (e.g., keeping the loss rate below 3% and maximum delay below 5 ms) while maintaining high bandwidth efficiency. The key is to leverage the economies of scale in the number of multiplexed voice flows. The relationship between the effective bandwidth and the number of multiplexed voice flows is useful in network planning for voice or multi-service VPNs. We discussed how the results can be used to design an algorithm for call admission control for voice VPNs based on measured statistics of the aggregate traffic and traffic specification from senders.

Further studies are required to determine how the choice of different codecs will affect the bandwidth usage and performance requirements of VoIP. For measurement based algorithms presented in Section 6, careful analysis is needed to choose an optimum window to estimate the mean and variance of the arrival rate. Another natural extension of this work is to investigate the interaction between TCP connections and VoIP traffic in the Diff-Serv architecture that we discuss, where the call admission control and WFQ we proposed are deployed at the boundary routers.

8 ACKNOWLEDGMENT

The authors would like to thank Bryan Lyles from Sprint Labs and Luca Salgarelli from Lucent Technology for sharing their expertise in this field. Special thanks are dedicated to Professor Anthony Joseph and other ICEBERG group members at Berkeley for the discussions and exchange of ideas that have been very enlightening and constructive.

9 REFERENCES

[1] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications,” *RFC 1889*.

[2] R. Cox and P. Kroon, “Low bit-rate speech coders for multimedia communications,” *IEEE Communications Magazine*, pp. 34-40, Vol. 34, No.2. December 1996.

[3] S. Fosse-Parisis Bolot, D. Towsley, “Adaptive FEC-Based Error Control for Interactive Audio in the Internet,” *IEEE Infocom’99*, New York, USA, March 1999.

[4] Integrated Services Working Group: http://www.ietf.org/html.charters/intserv-charter.html

[5] Differentiated Services Working Group: http://www.ietf.org/html.charters/diffserv-charter.html

[6] S. Blake et al., “An Architecture for Differentiated Services,” *RFC 2475*, December 1998.

[7] B. Gleeson et al., “A Framework for IP Based Virtual Private Networks,” Internet Draft, <draft-gleeson-vpn-framework-00.txt>, September 1998.

[8] H. Zhang, “Service Disciplines for Guaranteed Performance Service in Packet-Switching Networks,” *Proc. IEEE*, vol. 83, no. 10, October 1995.

[9] K. Nichols et al., “Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers,” *RFC 2474*, December 1998.

[10] K. Nichols, V. Jacobson, and L. Zhang, “A Two-bit Differentiated Services Architecture for the Internet,” *Internet draft, draft-nichols-diff-svc-arch-00.txt*, November, 1997.

[11] M. May, J.C. Bolot, A. Jean-Marie, C. Diot, “Simple Performance Models of Differentiated Services Schemes for the Internet,” *Infocom99*, New York, USA, March 1999.

[12] A. Floyd, and V. Jacobson, “Random Early Detection Gateways for Congestion Avoidance,” *IEEE/ACM Transactions on Networking*, vol.1, no. 4, pp. 397-413, August 1993.

[13] B. Braden et al., “Recommendation on Queue Management and Congestion Avoidance in the Internet,” *RFC 2309*, April 1998.

[14] D. Clark and J. Wroclawski, “An Approach to Service Allocation in the Internet,” *Internet draft, draft-clark-different-svc-alloc-00.txt*, July 1997.

[15] ITU-T Recommendation G. 114, “General Characteristics of International Telephone Connections and International Telephone Circuits: One-Way Transmission Time,” February 1996.

[16] H. Cramer, *Mathematical Methods of Statistics*, Princeton University Press, Princeton, N.J., 1946.

[17] S. Keshav, “REAL: A Network Simulator”, *Computer Science Department Technical Report 88/472*, UC Berkeley, 1988.